

A Completely Open-Source Finite Element Modeling Chain for Tubular Tissue-Engineered Constructs

Adrienne Madison and Mark A. Haidekker*

University of Georgia, College of Engineering, Athens, GA, USA

*Corresponding Author

Email: mhaidekk@uga.edu

Abstract

Finite-element modeling (FEM), well-established to predict the mechanical behavior of mechanical systems, enjoys growing popularity in the study of the biomechanical behavior of biological tissues. Contrary to mechanical systems, which can often be described analytically or with geometric primitives, biomedical objects require discretization of their often irregular shape. In noninvasive studies, imaging methods are used to obtain the shape. Frequently, a relationship between image intensity and biomechanical properties is assumed. Commercial FEM toolchains exist, but we failed to obtain a satisfactory discretization from simple phantom images. Driven by the application to image and characterize tissue-engineered blood vessels noninvasively, we sought to establish a completely open-source FEM toolchain. The open-source feature gives users the ability to modify and extend the code, and thus offers additional flexibility over commercial systems. We demonstrate that the combination of a custom module to discretize the geometry (meshing) combined with the open-source FEM solver *Tochnog* and free visualization software (namely, *Paraview* and *OpenDX*) completes an open-source FEM toolchain. We demonstrate its ability to analyze tubular phantoms modeled after tissue-engineered blood vessels and compare results to a commercial toolchain. We conclude that a fully open-source toolchain is feasible, but the critical element is the meshing module.

Keywords

Open-Source Software; Mesh Generation; Tissue Biomechanics; Vascular Grafts; Finite Element Modeling; FEM

Introduction

Finite-element modeling (FEM) has been intensely used in biomedical contexts to examine, simulate, and predict the material behavior and non-linear biomechanical properties of soft tissues through integration with medical imaging modalities [1]. Areas of application include the brain [2], lungs [3], and most importantly to us,

vascular stress analysis [4-7]. The underlying principle of FEM is to subdivide the object of interest into a large number of connected small volumes that are considered homogeneous. These small volumes -- the finite elements -- are considered to be rigidly connected, thus providing boundary conditions imposed upon the element by its neighbors (forces, displacement) and by the environment (pressure, fixed and immobile elements, or moving elements).

One key challenge in the application of FEM in the biomedical context is the extraction of the object of interest from a medical image (i.e., segmentation) and the conversion of the object into a connected set of finite elements (meshing) that not only represents the geometry of the object, but also the approximate local material properties in the tissue section that corresponds to the finite element. Only when this information is available -- the geometry, discretized into finite elements, and the associated material properties -- the response of the tissue to external mechanical loads and the pressure can be examined. Algorithms capable of constructing triangular and tetrahedral-shaped FEM meshes from medical images exist [8,9], yet neither study addresses the issue of segmentation. In fact, segmentation is usually considered a separate step, although one pre-processing toolkit [10] specializes in combining segmentation and meshing of 3D models. The algorithms implemented are also capable of material property assignments. Another related software toolkit reported in [11] provides the assignment of both material properties as well as boundary/loading conditions.

The driving application behind the toolchain presented in this manuscript is the recent development of fully biological tissue-engineered blood vessels [12,13]. Blood vessels that are grown from the patient's own cells must be tested exhaustively on an individual sample basis

because of the inter-sample variation. Our vision is to use an imaging method, such as optical coherence tomography (OCT), confocal imaging [14,15], or optical trans-illumination tomography [16], then use the resulting image data to generate a FEM mesh that allows the prediction of its biomechanical behavior when exposed to pulsatile blood pressure. Studies exist where FEM was applied on vascular entities, such as arteries and veins [4-7, 17-19]. Commonly, model geometries, based on measurements obtained from real vessels, were constructed with commercial CAD and FEM software. However, these studies only focus on the results obtained from the analysis, and not specifically on the processes to prepare 3D image data for analysis; detail information in those publications was accordingly sparse.

Several finite element solvers are available from different companies. We initially used a departmental license of Algor

(Algor, Inc., Pittsburgh, PA), a combination of FEM simulation software with limited visualization features. The generation and analysis of analytically-defined cylindrical models was straightforward, and no fundamental challenges were encountered. However, it was not possible to manually generate the complex and irregular shapes of a sample blood vessel. A separate software program was necessary to extract the mesh from the original image. Few such programs are offered, because engineering design often allows to describe mechanical objects in an analytical fashion or as a combination of geometrical primitives. One software program that allows to generate a mesh from medical images is Mimics (Materialise, Leuven, Belgium). We prepared a simple phantom (**Figure 1**) to test the ability of Mimics to generate a mesh that can be processed by Algor. In spite of extensive support from Materialise, we were not able to generate such a mesh. The Mimics meshes showed irregular element boundaries and shapes, and Mimics produced a very large number of nodes that could not be reduced and that caused Algor to fail. We concluded that the combination of Mimics and Algor would not meet our requirements. To avoid paying the license fees for additional software, we decided to attempt the design of a FEM chain based entirely of free open-source software for use in the analysis of tubular constructs.

Materials and Methods

Finite-element modeling of tissue engineered blood vessels is generally performed using some form of volumetric image obtained, for example, by computed

tomography (CT), magnetic resonance imaging (MRI), or -- on a smaller scale -- by optical coherence tomography (OCT) and involves four main steps. In the first step, the object of interest is separated from the image background (segmentation). The meshing step follows, during which the segmented object is subdivided into small elements, and boundary conditions and material properties are applied to the elements. The third step involves the actual finite element simulation of the time-variable behavior of the object under the established parameters. Once this is completed, the output of the simulation process is visualized or post-processed in the final step.

Phantom Generation

Two phantoms were used throughout this study that approximated pre-segmented blood vessels when scanned in air with computed tomography. Both phantoms were embedded in a 256 by 256 by 128 voxel matrix. Voxels were anisotropic with a size of approximately 23 microns in the x-y-plane and 92 microns in the axial (z) direction. The first phantom, shown in **Figure 1**, was modeled after an autologous vascular graft described by L'Heureux et al. [13]. The vascular graft has typical dimensions of 5.0×10^{-4} m (0.5 mm) thickness, 4.0×10^{-2} m (40 mm) length, and 4.0×10^{-3} m (4 mm) diameter, which is closely matched by our phantom, except that the length was truncated to approximately 1.2×10^{-2} m (12mm).

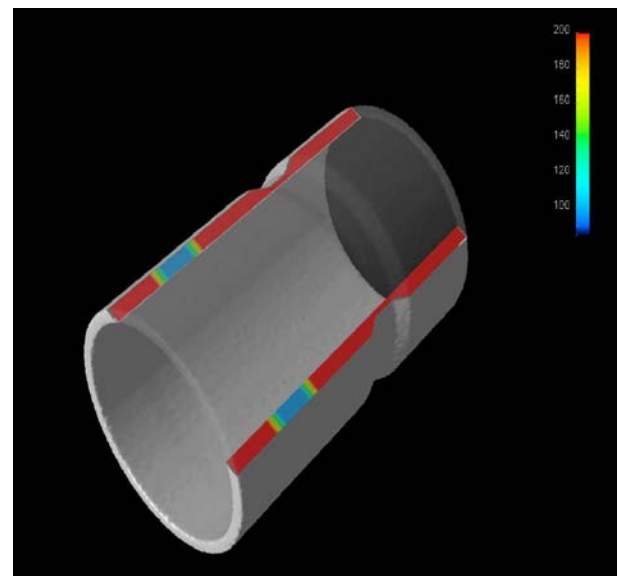


FIG. 1 3D RENDERING OF THE FIRST PHANTOM USED IN THIS STUDY. THE PHANTOM IS A TUBULAR OBJECT THAT IS WIDELY HOMOGENEOUS, BUT HAS TWO INHOMOGENEOUS REGIONS. THE FIRST REGION HAS A THINNER WALL, WHEREAS THE SECOND REGION HAS LOWER IMAGE VALUES. IN THIS RENDERING, THE OBJECT HAS BEEN CLIPPED, AND THE CUT SURFACES HAVE BEEN FALSE-COLORED TO HIGHLIGHT THE CHANGE IN DIAMETER AND CHANGE IN IMAGE VALUES.

The first phantom was widely homogeneous, but featured two annular inhomogeneities, one section where the wall thickness was reduced, and one section where the image intensity was lowered, while the wall thickness was held constant. The first section was created to simulate the fixation grooves (visible in **Figure 2**), and the second section represented incompletely fused tissue layers. This second section translates into different material groups. For comparison, the cross-section of a CT image of an actual tissue-engineered blood vessel is shown in **Figure 2**.

The second phantom was modeled after a fusiform aneurysm in a curved section of blood vessel. The outer dimensions, that is, 4.0×10^{-3} m diameter, 5.0×10^{-4} m wall thickness and 1.2×10^{-2} m length were kept similar to the first phantom, but a 15° bend was introduced, and the central region featured an elliptical expansion, with the inner circular region filled with a simulated plaque, i.e., a region with higher material stiffness. A rendering of the second phantom is shown in **Figure 3**.

Mesh Extraction Module

We decided to develop a relatively simple mesh extraction module based on the principle of radial probing rays [20] that seamlessly interfaces with the chosen open source FEM software, uses the near-cylindrical geometry, and may serve as a demonstration model for more advanced meshing modules.

The input of the module is assumed to be pre-segmented, that is, all background voxels have a zero value, and all voxels that belong to the vessel phantom have a nonzero value. In a first pass over the image, the centroid for each slice is determined and a straight line fitted into these centroids. This line is a first-order approximation of the lumen center capable of reflecting a tubular object that is tilted or slightly curved.

In the second pass over the sample, radial lines are emitted in each slice from the point where the central line intersects the slice (**Figure 4**). Starting from this point, image values are sampled along each ray. Once a nonzero image value is encountered for the first time, this point is marked as a node on the inner wall. Tracing continues until the image values drop to zero

again, and this point is marked as a node on the outer wall. Thus each ray provides one pair of nodes. Two adjoining rays together with their corresponding rays in the next slice provide eight nodes, and these eight nodes define one finite element. The relationship of the rays and nodes to one element is shown in **Figure 5**. At the same time, image values inside the approximated cuboid are averaged to provide a material index for the respective element. After user-selectable binning, the element can be assigned to a material group, and Young's modulus for each material group is user-specified.

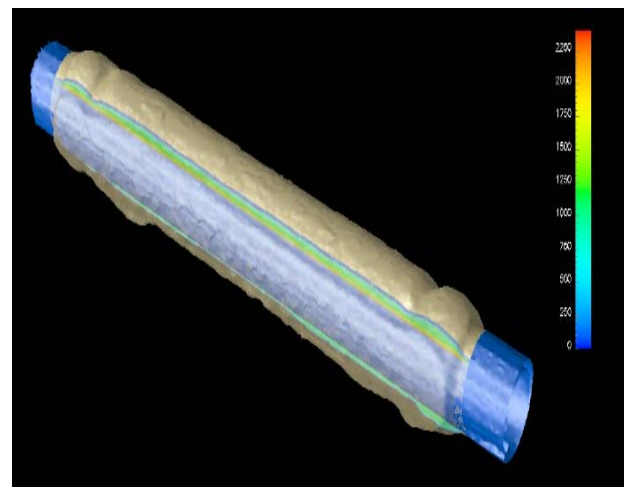


FIG. 2 3D RENDERING OF A TISSUE-ENGINEERED BLOOD VESSEL [13] OBTAINED BY COMPUTED TOMOGRAPHY. THE TISSUE LAYER (INDICATED IN AN OFF-YELLOW COLOR) IS GROWN ON A STEEL MANDREL (BLUE) WITH 4 MM OUTER DIAMETER. THE TISSUE COMPLETELY ENCLOSES THE MANDREL, BUT THE TISSUE WAS CLIPPED IN THIS IMAGE TO SHOW THE CROSS-SECTIONAL INTENSITY DISTRIBUTION AND THE THICKNESS IRREGULARITIES. THE FIXATION GROOVES SHOW PROMINENTLY AT EACH END OF THE TISSUE SECTION. NOTE THAT THE APPARENT TISSUE DENSITY INCREASE NEAR THE MANDREL IS AN ARTIFACT CAUSED BY PARTIAL-VOLUME EFFECTS.

In a third pass over the now-generated surface discretization, mesh refinement by interpolation is possible. When interpolation is selected, a user-defined number of nodes are interpolated between two adjoining rays by means of natural cubic splines.

The meshing module alternatively outputs a STL (stereo lithography) file for immediate visualization, or a Tochnog input/control file. The Tochnog file structure, together with an explanation of the required sections for the FEM software, is provided in the Appendix.

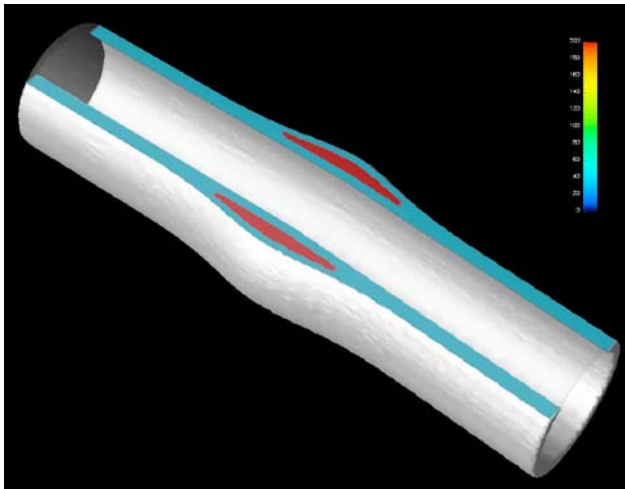


FIG. 3 3D RENDERING OF THE ANEURYSM MODEL FOR THE SECOND PHANTOM. THE MODEL REPRESENTS A FUSIFORM ANEURYSM ALONG A BENT BLOOD VESSEL WITH A 15° CURVE. THE FALSE-COLORED IMAGE VALUES REPRESENT THE MATERIAL ELASTICITY, IN THIS CASE, LIGHT BLUE FOR THE REGULAR VESSEL WALL AND RED FOR THE MORE RIGID PLAQUE. THE COLOR SCALE BAR IS THE SAME AS IN FIGURE 1.

FEM Software

A mature FEM solver that is available under a free open-source license exists. Tochnog, an implicit/explicit solver for a number of different problems, including linear and nonlinear solids, fluids and gases (Navier-Stokes), and for solving the wave equation, can be found at (tochnog.sourceforge.net). The output file generated by the mesh extraction module directly serves as a Tochnog input/control file, and only minimal editing (specifically, modification of the material definition) is needed. Control statements within the Tochnog input file determine its output format. At the end of the simulation run, the output file contains the node locations, node pressure, temperature, stress, strain, and velocity for each time step. Tochnog generates output files that are directly compatible with various post-processing and visualization packages.

Visualization

The visualization program GiD (CIMNE, Barcelona, Spain) is recommended by Tochnog. GiD is available without a license fee, but the no-cost version comes with a time limit. We examined both Paraview (Kitware, Inc., Clifton Park, NY) and OpenDX (www.opendx.org) as alternative visualization modules at the end of the FEM processing chain. Paraview accepts many input formats, and we used the VTK (visualization toolkit) format to exchange

data between Tochnog and Paraview. OpenDX requires a specific input format, which Tochnog can provide. In addition, the meshing module can produce an STL file, which is a tessellated description of the object's surface without any material properties. Many STL file viewers are available, and we used gmsh (<http://geuz.org/gmsh/>).

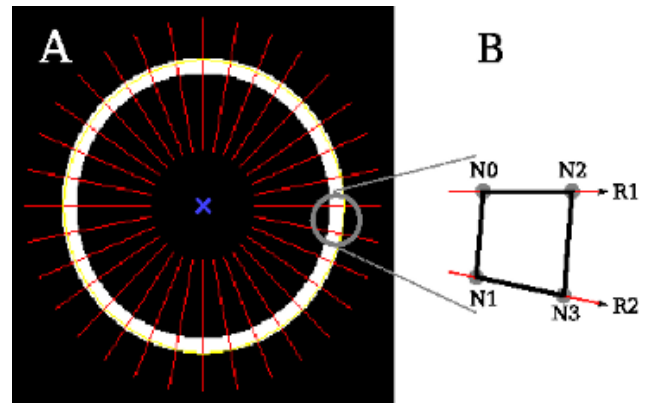


FIG. 4 EXTRACTION OF THE BOUNDARIES OF A CONVEX, TUBULAR OBJECT WITH PROBING RAYS (A). IN EACH SLICE, PROBING RAYS ARE EMITTED FROM THE CENTROID (BLUE X-MARK) AT REGULAR, ADJUSTABLE ANGULAR INTERVALS. IMAGE VALUES ARE SAMPLED ALONG THE PROBING RAYS. WHEN A PRE-SELECTED THRESHOLD IS FIRST EXCEEDED, AN INTERSECTION (NODE) OF THE RAY WITH THE INNER WALL IS RECORDED. ONCE THE IMAGE VALUES DROP BELOW THE THRESHOLD AGAIN ALONG THE RAY, THE INTERSECTION OF THE RAY WITH THE OUTER WALL IS RECORDED. TWO SUBSEQUENT RAYS (R1 AND R2) THEREFORE DEFINE A QUADRILATERAL, WHICH IS ONE FACE OF AN ELEMENT. A MAGNIFIED SECTION (B) SHOWS THE NODES. THE NUMBERING OF THE NODES CORRESPONDS TO FIGURE 5, AND NODES NEED TO BE ORDERED AS INDICATED BY THE NODE NUMBERS.

Results

Proof of Principle

To test the overall FEM chain from the initial volumetric image to the final visualization, we assumed that our phantom model (Figure 1) was a thin-walled, relatively elastic blood vessel being subjected to an internal pressure that corresponds to blood pressure acting on the inner lumen of the vessel. We also assumed that the ends of the vessel would be fixated (fixation grooves in Figure 2). Initially, we used 60 probing rays per slice and one set of rays every slice for a total of 15,360 nodes and 7,620 elements. The ray values were selected such that the division of the number of rays by the 360 degrees in a circle would result in a whole number multiple of 360.

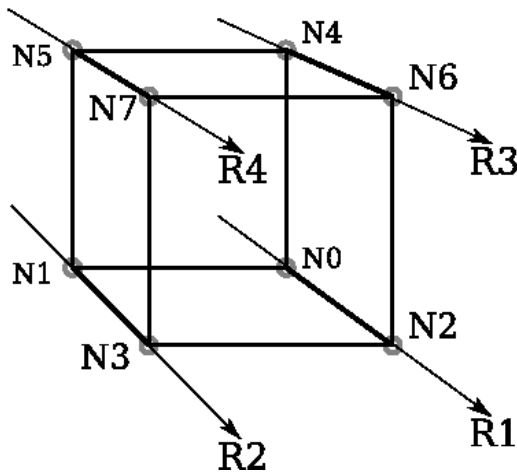


FIG. 5 RELATIONSHIP OF THE PROBING RAYS (R1 THROUGH R4) TO THE NODES AND FACES OF AN ELEMENT. THE LOWER TWO RAYS (R1 AND R2) BELONG TO SLICE Z , WHEREAS THE UPPER TWO RAYS (R3 AND R4) BELONG TO THE SUBSEQUENT SLICE AT $Z+\Delta Z$. FROM THE OBSERVER'S POINT OF VIEW, RAYS ARE PROCESSED FROM RIGHT TO LEFT. THE NODES (INDICATED BY GRAY CIRCLES) ARE ARRANGED IN A ZIGZAG PATTERN WHERE THE FIRST TWO NODES (N0 AND N1) LIE ON THE INSIDE WALL, AND THE NEXT TWO NODES (N2 AND N3) LIE ON THE OUTSIDE WALL, WHEREBY THE CONNECTING VECTORS $N0 \rightarrow N1$ AND $N2 \rightarrow N3$ BOTH POINT RIGHT-TO-LEFT. THE SAME ORIENTATION IS USED FOR THE NODES IN THE UPPER SLICE, N4 THROUGH N7.

Using an in-plane pixel size of $2.3 \times 10^{-5}m$, we subjected the internal nodes to a pressure of -16 kPa (approximately 120 mmHg). The negative sign indicates outward-directed pressure. Each end of the tube was fixed by labeling them as boundary nodes and restricting movement in the x -, y -, and z -directions. We selected a Poisson ratio of 0.45, and a density of 945 kg/m^3 . Based on the average intensity values calculated, five material groups were generated possessing elastic moduli values of 55, 60, 75, 85, and 100 MPa which were assigned and chosen to make the construct relatively elastic. All of the mechanical property values were selected such that our model could be characterized as having a rubber-like material composition.

The visual results of subjecting our phantom model to our modeling chain, created with GiD, are presented in **Figure 6**. The images depict the generated mesh onto the 3D tubular model, the classification of material groups present, as well as an exaggeration of the deformation behavior resulting from pressure being applied to the inner wall.

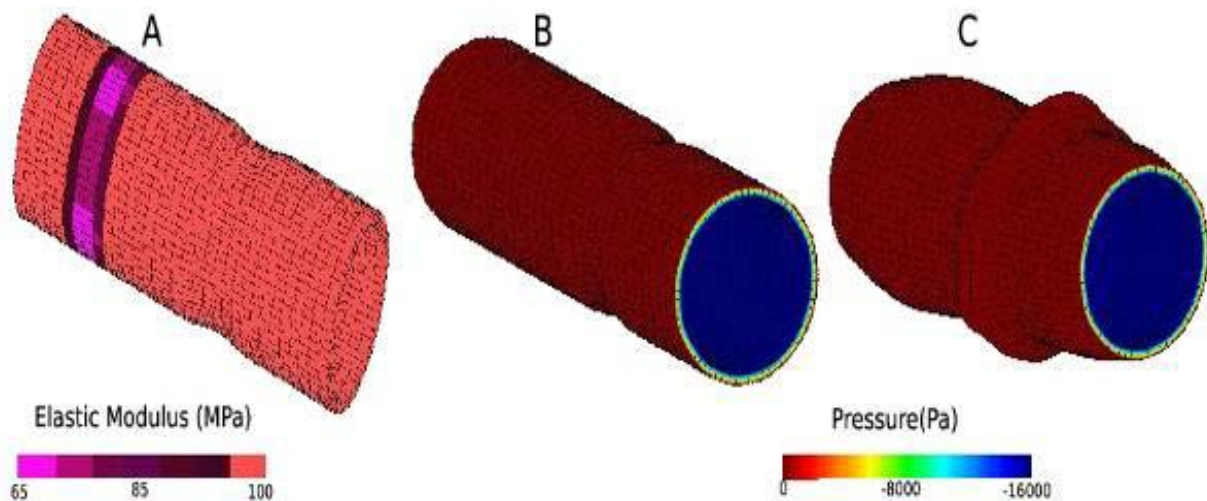


FIG. 6 GID VISUALIZATION RESULTS OF PHANTOM MODEL SUBJECTED TO HOMOGENEOUS INTERNAL PRESSURE. THE PHANTOM IS DECOMPOSED INTO 15,360 NODES AND 7,620 ELEMENTS. THE CIRCULAR SECTION WHERE THE WALL IS THINNER AS WELL AS THE CIRCULAR SECTION WHERE IMAGE INTENSITY IS LOWER BECOME VISIBLE. REDUCED IMAGE INTENSITY IS REFLECTED IN MULTIPLE MATERIAL GROUPS (A). THE INTERNAL PRESSURE EXPANSION AND DISTRIBUTION BEHAVIORS OF THE THIN WALL AREA, LOWER INTENSITY AREA, AND ENDS OF TUBE ARE HIGHLIGHTED (B). THE EXAGGERATED SHAPE DEFORMATION RESULTING FROM THE INTERNAL PRESSURE EXPANSION IS ALSO SHOWN (C).

Mesh Convergence

We examined possible convergence behavior by varying (1) the number of nodal interpolation points between rays and (2) the ray distance in the z- (axial) direction. The material stress values from each analysis were chosen as the basis of the convergence observations. The results are presented in **Table 1**. In the case of (1), we used spline interpolation to generate additional nodes without changing the nodes of the non-interpolated reference mesh. For these experiments, 18 probing rays per slice were implemented because the angular increments of 20° provide adequate spacing for the addition of interpolated points between each pair of rays. Analyses were conducted on the tube that resulted in the addition of 0-8 nodes in between pairs of rays at each slice. The maximum stress values obtained at each interpolation level are similar, with maximum deviations of -2% and +3% from the mean value of 19.4 kPa. The axial anisotropy is the product of voxel anisotropy and the number of slices skipped for the mesh generation. For the geometry defined for the phantom, voxels are 4 times longer in the axial direction than in the x-y-plane. In the case of varying ray distance in the axial direction, two analyses were conducted at 18 rays without the addition of interpolated nodal points: (a) 18 rays per slice with one set of rays every four slices and (b) 18 rays every four slices, resulting in axial anisotropies of 4 and 16 respectively. We observed that number of nodes and the number of elements increase proportionally with the number of interpolated points; however, they decrease proportionally as the axial anisotropy increases. While the number of material groups are not affected by the number of rays and nodal and elemental amounts, increasing the axial anisotropy also decreases the number of material groups present. The internal wall pressure and expansion behavior remains unaffected by changes in nodal and elemental amounts. Lastly, the maximum stress values obtained at the higher axial anisotropy of 16 and in our proof-of-principle experiment using 60 rays (19.5 kPa) align with the values with presented in the table obtained using both a lower anisotropy of 4 and lower number of rays.

An out-of-the-box installation of Tochnog does not have multiprocessor capabilities, but a typical file with approximately 16,000 nodes takes only a few minutes to compute on a single CPU core. The amount of computational time required increases

approximately linearly with increasing nodal and elemental amounts and decreases accordingly when the axial anisotropy increases. However, Tochnog can be linked with libraries that are multi-thread-capable, and execution time is reduced correspondingly.

Evaluation of the Simulated Aneurysm Geometry

In addition to the cylindrical geometry of the first phantom, we examined the performance of our meshing module in cases of more complex tubular geometries. Aneurysms are balloon-like bulges that occur in blood vessels as the inner blood vessel wall weakens and eventually tears as a result of high blood pressure. Blood, and in some instances plaque, begins to pool in localized positions between these weakened vessel layers. Continued growth or expansion of the bulge increases the potential for the vessel to rupture, leading to a host of other complications including hemorrhaging, stroke, and in extreme cases, death.

We decided to simulate a fusiform aneurysm in which a bulging deformation is visible on both sides of the vessel **Figures 3 and 7A**. In addition to its expanded section in the center with two different material properties, there is a height-dependent offset of the vessel geometry in the x-direction, resulting in a 15° angle between the medial axes of the two ends. The purpose of this kink is to prove that the meshing algorithm can accurately capture objects that are not fully cylindrical. With 60 probing rays per slice, the model is composed of a total of 15,360 nodes and 7,620 near-cuboid elements. The in-plane pixel size, pressure, Poisson ratio and density values were the same as outlined in Section 3.1. Based on the average intensity values calculated, eleven material groups were generated possessing elastic moduli values of 100, 105, 110, 115, 120, 125, 130, 135, 140, 145, and 150 Mpa. Elastic moduli with larger values are expected to be assigned within the bulging areas of the vessel. Different material groups occur when the element encloses different amounts of the two materials, an effect similar to partial-volume artifacts in computed tomography.

Figure 7B displays the material stress obtained from the Tochnog analysis. As expected, the highest material stress values are observed at the weakest areas of the inner vessel walls, where the bulging between layers is first observed. The areas of the vessel perpendicular to the bulges are believed to also

TABLE 1 EFFECTS OF AXIAL AND RADIAL MESH REFINEMENT ON THE TOTAL NUMBER OF NODES AND ELEMENTS, MAXIMUM STRESS VALUES, AND NUMBER OF MATERIAL GROUPS, AND FEM SOLVER EXECUTION TIME FOR THE TUBULAR PHANTOM.

Number of Rays	Axial Anisotropy	Interpolation	Nodes	Elements	Material Groups	Maximum Stress (kPa)	Time (min:sec)
18	4	0	4608	2286	5	19.0	0:22
18	4	1	9216	4572	5	19.1	0:45
18	4	2	13824	6858	5	19.2	1:10
18	4	3	18432	9144	5	19.8	1:32
18	4	4	23040	11430	5	19.0	1:55
18	4	5	27648	13716	5	19.2	2:24
18	4	6	32256	16002	5	20.0	2:48
18	4	7	36864	18288	5	19.6	3:17
18	4	8	41472	20574	5	19.9	3:45
18	16	0	1152	558	2	19.6	0:13

be subjected to increased stress due to the shape deformation (bulging) under high pressure and difference in material property values in comparison to the bulge.

Visualization Software Comparison

The images presented in **Figure 6** were constructed using GiD; however to adhere to our goal of presenting a completely open-source modeling chain, we compared the visualizations from GiD with those from Paraview and OpenDX. Paraview was found to provide visualizations comparable to GiD, whereby out-of-the-box visualization schemes provided immediate results with a low level of user input. A Paraview rendering example is presented in **Figure 7**. Conversely, OpenDX allowed to design complex visualizations with high flexibility. One sophisticated example is shown in **Figure 8**. However, OpenDX requires the generation of a visual program, and experience with OpenDX is a prerequisite. A unique feature of GiD is the ability to exaggerate deformation (prominently visible in **Figure 6C**). A comparable feature was not found in Paraview.

Lastly, we wanted to compare the performance of our meshing module to the mesh previously obtained by the closed-source medical image pre-processing software. We used Mimics to obtain the 3D mesh of our phantom model shown in **Figure 1**. This mesh was then imported into the commercial FEM software Algor for visualization purposes. The resulting mesh consisted of a mix of approximately 300,000 hexagonal, wedge, pyramid, and tetrahedral shaped elements and comprised approximately 100,000 nodes

and 13,000 surfaces. **Figure 9** displays the visual results obtained from the Mimics meshing process. The simulation performed with Algor did not converge and did not produce any output.

Discussion

In this study, we presented a fully open-source toolchain for finite-element modeling. A number of reasons make such a toolchain an attractive alternative to commercial software packages. First and foremost, no license fees are incurred, and FEM analysis becomes possible on a low budget. Furthermore, the usability of the software for a specific purpose can be examined without obligation. The second fundamental advantage lies in the open-source nature of the software, which means that the underlying program code can be examined or modified. In a classroom setting, where FEM is often taught with black-box software under an academic license, students can use open-source software to examine the numerical aspects of solving discrete partial differential equations. In a research setting, the exact algorithm that leads to a specific result can be determined, potential weaknesses identified, and the code amended. Access to the algorithm is particularly important when critical nonlinear cases are examined, such as turbulence or fracture. On the other hand, open-source software generally has no organized support. Rather, this type of software relies on community support, and software-related questions are usually resolved by peers in Internet forums. Software development is also driven by community efforts, although sometimes (as in the example of Paraview) a company supports development. The centerpiece of our

toolchain, the FEM solver Tochnog, is offered both as a free version and as a commercial version with paid support and development (Tochnog Professional, Feat, The Netherlands).

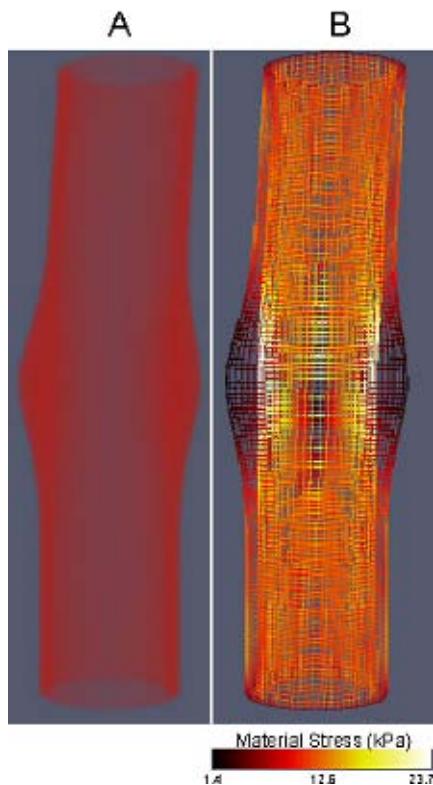


FIG. 7 PARAVIEW VISUALIZATION RESULTS OF THE FUSIFORM ANEURYSM PHANTOM MODEL SUBJECTED TO HOMOGENEOUS INTERNAL PRESSURE. COMPOSED OF 15,360 NODES AND 7,620 ELEMENTS, THE VOLUMETRIC GEOMETRY IS SHOWN (A) ALONG WITH THE INTERNAL AND EXTERNAL STRESS DISTRIBUTION THROUGHOUT THE VESSEL (B).

If sufficient demand exists, community-driven software packages of similar functionality can frequently be found. The visualization toolkit VTK, for example, has led to the development of Paraview, VisIt (Lawrence Livermore National Laboratory, Livermore, CA), and Slicer (www.slicer.org), to name a few examples. Alternatives to Tochnog exist as well, for example, FreeFEM (www.freefem.org) and Elmer (elmer.sourceforge.net). However, we found no free alternative to Mimics to create a mesh from a 3D medical image. This gap prompted us to develop a specialized meshing program for our research application of vascular constructs, and the implications of this gap are discussed in detail below.

Meshing Module

The most crucial step in all FEM chains that process medical images is the actual surface or volume parametrization. In simple terms, one could describe this step as converting a stack of pixels into

an ordered set of nodes and elements. This step consists of two successive parts: Image segmentation and the actual surface parametrization. Interestingly, few commercial software packages and no open-source packages exist that perform this task. Mimics by Materialise is arguably the most widely used mesh generation package for medical images. Our experiments with Mimics led to meshes with a large number of irregular shapes, jagged edges and sharp drop off points. We observed that downstream software had difficulties processing such a mesh. Since we were unable to find the cause of those difficulties in spite of extensive help from Materialise support, we decided to favor the simplest possible geometry for our application and develop our own mesh extraction software with the key difference that, unlike Mimics, our software will be restricted to a limited set of geometries.

A second notable difference to Mimics is the ability of our module to automatically assign material parameters that correspond to varying image intensity values. Such a function does not exist in Mimics, which assumes a single homogeneous material. In addition, the adjustment of probing ray density, spline-based surface smoothing, and axial anisotropy adjustment features of our meshing module do not have a known correspondence in Mimics. These examples illustrate a typical trade-off between software aimed at solving as broad a range of problems as possible (Mimics) and software aimed at solving specialized cases (our meshing module).

In comparison with other FEM software chains, the uniqueness of our approach lies in the completely open nature of all steps of mesh generation, solving of the equations, and visualization. Other published partly open-source chains require the generated model to be imported and analyzed with commercial closed-source software [10, 11], or, in other cases, the meshing algorithms examined require the use of commercial software to both pre-process the image slices into a 3D model and undergo FEM analysis and do not appear to be open-source [8, 9]. In addition, most of these have only been implemented on anatomical bone structures, which are by nature more easily segmented.

Our meshing software extracts geometry from the image and applies a highly regular grid of nodes to the inner and outer wall. In addition, the software is capable of applying material properties and boundary/loading conditions to the model. Using the

probing ray principle, the module is able to automatically generate finite cuboids, trapezoids, or frustums in a Tochnog-ready input file. Through calculation of average intensities, we were able to distinguish between inhomogeneities along the model and assign unique material definitions on an element-by-element basis under the assumption that a relationship between image intensity and material properties exists [21-24]. This assumption has also been implemented in other FEM analyses of medical images [3, 10]. This relationship is often empirical and highly dependent on the imaging modality used. In some examples, notably bone imaged by computed tomography, a strict relationship between the CT value and bone mineral density exists. Optical modalities can relate scattering to the presence of collagen [15], which is fundamental to tissue elasticity. In other cases (e.g., T1 and T2 relaxation in MRI), the relationship is more tenuous and needs to be established beforehand in separate studies. For the proof of principle that is the focus of this study, we used somewhat arbitrary values that are common for many elastomers, and we are aware that the precise model of a tissue-engineered blood vessel requires additional research, for example, a study with optical tomography [16], where the blood vessel is subjected to pressure and its expansion measured.

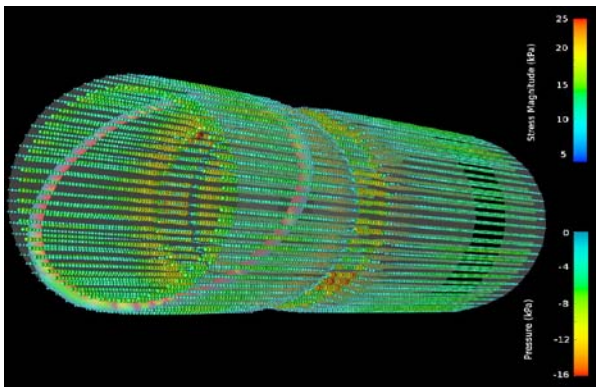


FIG. 8 EXAMPLE 3D RENDERING OF THE FEM SIMULATION RESULTS WITH OPENDX. FOR THIS EXAMPLE, THE INNER AND OUTER SURFACE WERE RENDERED AS GRAY, SEMI-TRANSPARENT TUBES. THE MAGNITUDE OF THE SHEAR STRESS TENSOR WAS SUPERIMPOSED FOR EACH NODE AS A GLYPH, I.E., A SMALL SPHERE WHERE THE SIZE IS PROPORTIONAL TO THE STRESS MAGNITUDE. THE GLYPHS ARE FALSE-COLORED WITH THE MAGNITUDE FOR IMPROVED VISUAL PERCEPTION AND THE VALUES REPRESENTING THESE COLORS ARE DISPLAYED IN THE UPPER COLORBAR. IN ADDITION, A SLANTED RING WAS PLACED INSIDE THE VESSEL WALL (NEAR THE LEFT END OF THE TUBE) THAT DISPLAYS THE MAGNITUDE OF THE PRESSURE BY USING THE LOWER COLORBAR (NEGATIVE VALUES INDICATE OUTWARD-DIRECTED PRESSURE).

In its present form, our meshing module is limited to approximately cylindrical, tubular objects. Concavities, for example, saccular aneurysms, would not be captured correctly by our module. Furthermore, the relatively coarse subdivision into one single element across the wall may cause some artificial rigidity. However, Tochnog performs automatic mesh refinement based on the residuals of the governing equations. Furthermore, our algorithm can be refined in a very straightforward manner with multiple thresholds. Presently, a binary image is assumed with one object of approximate radial homogeneity. If this assumption leads to an unacceptable simplification, a node along the ray can be created whenever one of multiple thresholds is crossed. This process may generate new shapes, most notably, triangular prisms and hexahedrons, for which different element definitions exist in Tochnog.

A consistent observation of radial inhomogeneity was made, which becomes most prominently visible in **Figure 6A**. More specifically, larger stresses and larger apparent deformations were observed at angles of 45°, 135°, 225°, and 315° with respect to the x - and y -axes than coincident with the axes. This inhomogeneity is the consequence of the discretization of a cylindrical object on a polar grid in the phantom bitmap image: Parallel to any axis, the phantom is exactly 10 pixels (2.3×10^{-4} m) thick, whereas rays at odd multiples of 45° detect the inner and outer boundaries approximately 8 pixels (1.84×10^{-4} m) apart. This apparent inhomogeneity is correctly recognized by the meshing algorithm and correctly visualized by the toolchain. However, it should be noted that this apparent inhomogeneity indicates the critical role of the object discretization and points at a potential source of error for all FEM toolchains when objects are extracted from medical images with low resolution.

In spite of its relative inflexibility, we believe that the meshing module is useful in its source form, because it demonstrates in detail how a Tochnog input file is composed. Therefore, the meshing module may readily be modified with other extraction algorithms (such as, e.g., the marching cubes algorithm) to accommodate a larger variety of shapes. For example, a simple extension towards spherical objects can be envisioned if the probing rays are emitted radially in all directions from the centroid. We did not further pursue this path, because the ray-based algorithm proved to be sufficient for our application, and because generalized algorithms based on the

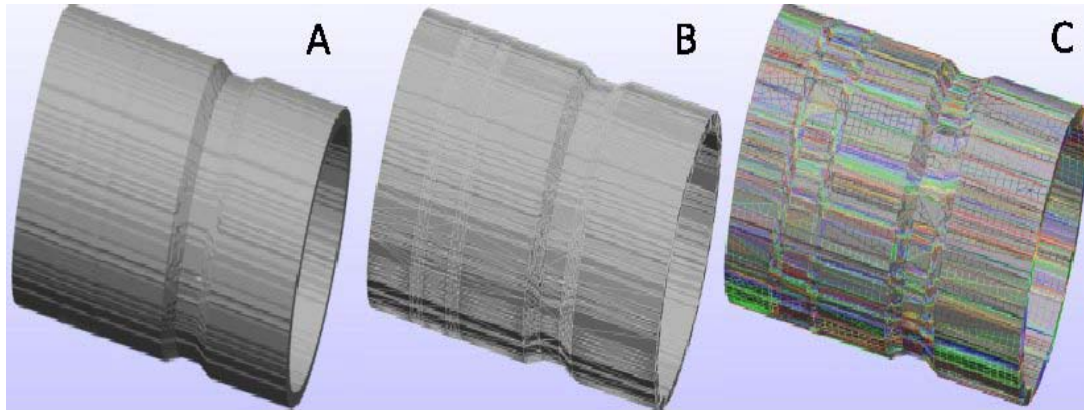


FIG. 9 VISUALIZATION RESULTS OF THE FIRST PHANTOM MODEL SUBJECTED TO COMMERCIAL IMAGE PRE-PROCESSING SOFTWARE THE GEOMETRIC RENDERING (A), THE MESHED MODEL (B), AND THE COLORED IDENTIFICATION OF ALL THE SURFACES AND ELEMENTS THAT THE MODEL IS COMPRISED OF (C) ARE HIGHLIGHTED.

marching-cubes algorithm exist [25-28]. By nature, familiarity with computer programming and the C/C++ language is helpful in understanding the algorithms and methods employed in the meshing module when a modification of our meshing algorithm is desired.

Visualization Software Options

Both of the open-source visualization software options evaluated are fully capable of providing suitable visualizations of FEM output data. Paraview is similarly intuitive as its commercial competitor, GiD. However, Paraview allowed us to obtain visualizations of the magnitude for stress, strain, and deformation values. This feature was not found in GiD; vector values are presented and must be calculated in order to obtain magnitude values. Conversely, Paraview does not allow us to visualize the varying material groups along the inhomogeneous section containing lower image intensity values. In addition, we were unable to visualize the exaggerated expansion behavior of the vessel when subjected to internal pressure as presented in **Figure 6**. OpenDX is advantageous in the sense that it allowed us to superimpose multiple aspects of the simulation onto one image. For instance, in the visualization presented in **Figure 8**, we were able to view the volumetric rendering of the tube (in light gray), the stress magnitude (as points in space), the pressure distribution throughout the tube (diagonal circle at left end of tube), and the inhomogeneous section of the tube containing lower image intensity values (black). Tochnog output files created for OpenDX are typically very large, because

all data for each point in time is recorded for processing in OpenDX. To extract specific data from the Tochnog output file, for example, the shear stress magnitude, specific visual programs must be created. Unlike Paraview and GiD, which are very intuitive, creation of a visualization with OpenDX requires experience with OpenDX. The key advantage of OpenDX is the unmatched flexibility with which visualizations can be designed.

Conclusion

The open-source philosophy allows the presented FEM chain to be available for use, modifications, improvements, and free distribution by our biomedical research peers. Our proposed modeling chain exhibits versatility as it can be applied to any tubular biomedical object that has been subjected to biomedical imaging, including medical device tubing, stents, or long bones. The meshing module is our contribution to the open-source community, and it can be freely downloaded at <http://haidekker.org/cimage/>. In this study, our module was applied only to phantom models of tubular constructs, but we have provided the framework which allows us to subject tissue-engineered vascular grafts to FEM analysis to examine biomechanical behavior. Future research will revolve around continuing to apply our modeling chain to objects of varying geometrical shapes and types of materials to test the specificity and efficiency of the probing ray principle.

ACKNOWLEDGEMENTS

Funding for this study was in part provided by the National Institutes of Health, grant 1R21 HL081308.

We thank Dr. Nicolas L'Heureux (Cytograft Inc., Novato, CA) for providing us with the tissue-engineered blood vessel sample that is shown in **Figure 2**.

Appendix

Interface between Meshing Module and Tochnog

Summary of key elements in the Tochnog control file: any meshing program needs to produce a file that contains the components described in this section.

- The header section. The header contains general control elements, for example, the number of spatial dimensions, and the simulated values we want to observe. In our case, we chose to observe the velocity, stresses, the total and elastic strains of the 3-D tubular construct when subjected to homogeneous pressure.

- The nodes section. This section lists all vertices (nodes) and their spatial location. Therefore, this section describes the geometry of the object. Nodes must be shared between adjoining elements. An example of lines that describe a group of nodes follows:

```
node 0    4.000000  0.000000  0.000000
node 1    3.695518  1.530734  0.000000
node 2    5.000000  0.000000  0.000000
node 3    4.619398  1.913417  0.000000
node 4    4.000000  0.000000  1.000000
node 5    3.695518  1.530734  1.000000
node 6    5.000000  0.000000  1.000000
node 7    4.619398  1.913417  1.000000
```

- Nodes are numbered consecutively (the number after the keyword "node"), and the spatial coordinates in a Cartesian system follow. It can be seen in this example that the first four nodes belong to the lowest slice at $z=0$, and the next four nodes belong to the slice at $z=1$.

- The elements section. In this section, nodes are grouped to form a cuboid element. Several element geometries are available in the selected FEM software, but we made use of a cube, which is referred to as *hex8* element in the software's terminology. A sample element definition follows:

```
element 1 -hex8  1 8 3 9 5 10 7 11
element 2 -hex8  8 12 9 13 10 14 11 15
element 3 -hex8  12 16 13 17 14 18 15 19
```

The number after the keyword "element" is the element number, used in material assignments. The instruction "-hex8" indicates the geometry, and the following eight integer numbers are the numbers of the nodes for this element. It can be seen that nodes are shared between adjoining elements. The order of the nodes is crucial. **Figure 5** shows the required order of the nodes.

- The element grouping section. In this section, each element is assigned to one material group. In one extreme case, only one material group exists, and each element is assigned to element group 0. In the other extreme case, variability is so high that each element has its own material group. Depending on material property binning, the number of material groups can be reduced substantially.
- The material properties section. In this section, material properties (among them, Young's modulus, Poisson elasticity, and material density) are defined. This is also the section where nonlinear material properties are introduced in the FEM software.
- The node boundary condition definitions. In this section nodes may be subjected to boundary constraints. In the special case that is considered in this study, all inner-wall nodes are subjected to a constant pressure, and all nodes in the lowest and uppermost slice are held fixed in space.
- The control section. In this section, the evolution in time of the simulation can be controlled. Factors include time steps, iteration limits, time intervals after which a snapshot is saved, and a control parameter of how the FEM software may subdivide the cubes if they become too inhomogeneous.

REFERENCES

- [1] D. J. Hawkes, D.Barratt, J.M. Blackall, C. Chan, P.J. Edwards, K. Rhode, G.P. Penney, J.McClelland, and D.L.G. Hill, *Medical Image Analysis* 9, 2 (2005).
- [2] A. Hagemann, K. Rohr, and H.S. Stiehl, *Medical Image Analysis* 6, 4 (2002).

- [3] T.A. Sundaram and J.C. Gee, *Medical Image Analysis* 9, 6 (2005).
- [4] J. Al-Sukhun, C. Lindqvist, N. Ashammakhi, and H. Penttilä, *Br. J. Oral Maxillofac. Surg.* 45, 2 (2007).
- [5] C.J. Beller, M.R. Labrosse, M.J. Thubrikar, and F. Robicsek, *Circulation* 109, 6 (2004).
- [6] C.J. Beller, M.R. Labrosse, M.J. Thubrikar, G.Szabo, F. Robicsek, and S.Hagl, *Eur. J.Cardiothorac. Surg.* 27, 2 (2005).
- [7] D.N. Ghista, A.S. Kobayashi, and N.Davids, *Computers in Biology and Medicine* 5, (1975).
- [8] G.H. Kwon, S.W. Chae, and K.J. Lee, *Computers and Structures* 81, (2003).
- [9] Z. Yu, M.J. Holst, and J. A. McCammon, *Finite Elements in Analysis and Design* 44, 11 (2008).
- [10] C.K. Chui, Z. Wang, J. Zhang, J.S.K. Ong, L. Bian, J.C.M. Teo, C.H. Yan, S.H. Ong, S.C. Wang, H.K. Wong, and S.H. Teoh, *Advances in Engineering Software* 40, 3 (2009).
- [11] N.M. Grosland, K.H. Shivanna, V.A. Magnotta, N.A. Kallemeyn, N.A. DeVries, S.C. Tadeipalli, and C. Lisle, *Computer Methods and Programs in Biomedicine* 94, 1 (2009).
- [12] N. L'Heureux, S. Pâquet, R. Labbé, L. Germain, and F.A. Auger, *The FASEB Journal* 12, 1 (1998).
- [13] N. L'Heureux, N. Dusserre, G. Konig, B. Victor, P. Keire, T.N. Wight, N. A. F. Chronos, A.E. Kyles, C.R. Gregory, and G. Hoyt, *Nature medicine* 12, 3 (2006).
- [14] J.T. LaCroix, J. Xia, and M.A. Haidekker, *Annals of Biomedical Engineering* 37, 7 (2009).
- [15] J.T. LaCroix, and M.A. Haidekker, *BMC medical imaging* 9, 1 (2009).
- [16] H.M. Huang, J. Xia, and M.A. Haidekker, *Annals of Biomedical Engineering* 36, 10 (2008).
- [17] C. Lally, F. Dolan, and P. J. Prendergast, *Journal of Biomechanics* 38, 8 (2005).
- [18] P. J. Prendergast, C. Lally, S. Daly, A. J. Reid, T. C. Lee, D. Quinn, and F. Dolan, *J. Biomech. Eng.* 125, 5 (2003).
- [19] W. Wu, W.Q. Wang, D.Z. Yang, and M. Qi, *Journal of Biomechanics* 40, 11 (2007).
- [20] M. Haidekker, R. Andresen, C. Evertsz, D. Banzer, and H. Peitgen, *The British Journal of Radiology* 70, 834 (1997).
- [21] L. Allard, G. Cloutier, and L. Durand, *Acoustical Imaging* 23, (1997).
- [22] Z. Guo and A. Fenster, *Ultrasound in Medicine and Biology* 22, 8 (1996).
- [23] Z. Guo, L. G. Durand, L. Allard, G. Cloutier, and A. Fenster, *Journal of Vascular Surgery* 27, 4 (1998).
- [24] G. Cloutier, Z. Qin, D. Garcia, G. Soulez, V. Oliva, and L. G. Durand, *Ultrasound in Medicine and Biology* 26, 9 (2000).
- [25] W. Lorensen, and H. Cline, *Computer Graphics* 21, 4 (1987).
- [26] T. Elvins, *ACM SIGGRAPH Computer Graphics* 26, 3 (1992).
- [27] C. Montani, R. Scateni, and R. Scopigno, *IEEE Conference Visualization*, 1994.
- [28] T. Nishimara, and T. Fugimoto, *Systems and Computers in Japan* 25, 3 (1994).